Remarks

Applicants respectfully request reconsideration of the present U.S. Patent application as amended herein. Claims 1, 2, 5, 6, 8-11, 13, 15-18 and 23 have been amended. No claims have been added or canceled. Thus, claims 1-27 are pending.

CLAIM REJECTIONS – 35 U.S.C. § 103(a)

Claims 1-27 were rejected as being unpatentable over U.S. Patent No. 6,058,491 issued to Bossen, et al. (*Bossen*) in view of U.S. Patent No. 6,023,772 issued to Fleming, et al. (*Fleming*). For at least the reasons set forth below, Applicants submit that claims 1-27 are not rendered obvious by *Bossen* and *Fleming*.

Claim 1 recites:

> storing a register architectural state of a processor corresponding to a first checkpoint;
>
> storing non-deterministic memory access events as associated data that occur subsequent to the storage of the first checkpoint;
>
> determining whether an processing error has occurred subsequent to the storage of the first checkpoint; and
>
> restoring the register architectural state of the processor corresponding to the first checkpoint and re-executing the non-deterministic memory access events if a processing error is detected.

Similarly, claim 15 recites:

> means for storing a register architectural state of a processor corresponding to a first checkpoint;
>
> means for storing non-deterministic memory access events as associated data that occur subsequent to the storage of the first checkpoint;
>
> means for determining whether an processing error has occurred subsequent to the storage of the first checkpoint; and
>
> means for restoring the register architectural state of the processor corresponding to the first checkpoint and re-executing the non-deterministic memory access events if a processing error is detected.

Thus, Applicants claim storing a register architectural state corresponding to a checkpoint and logging of non-deterministic memory access events subsequent to the checkpoint. If an error occurs after the checkpoint, the architectural register state is restored from the checkpoint and the non-deterministic memory access events are re-executed.

Bossen discloses storing the internal state information of one of the processors at each checkpoint. See col. 5, lines 52-55. *Bossen* discloses fault detection before the memory commit point. The processor results of the two processes are compared ***after each macro-instruction***. See col. 7, lines 1-2. When an error is detected, an error-free architectural state is achieved by rolling back the architectural state stored in the register file using the lookback state to obtain the architectural state from two instructions back. See col. 6, lines 38-41; col. 7, lines 19-32 and Figure 6.

Because, according to *Bossen*, fault detection occurs before the memory commit point, logging of the non-deterministic memory access events is unnecessary. Therefore, *Bossen* teaches away from the checkpoints as claimed in claims 1 and 15.

Fleming discloses storage of non-deterministic event information as incurring substantial processing overhead and a problem to be overcome. See col. 2, liens 62-65. Further, *Fleming* discloses roll-back as generally not attractive. See. col. 3, lines 3-5. Therefore, neither *Fleming* nor *Bossen* teach or suggest logging of non-deterministic memory access events.

Claims 2-14 depend from claim 1. Claims 16 and 17 depend from claim 15. Because dependent claims include the limitations of the claims from which they depend, Applicants submit that claims 2-14, 16 and 17 are not rendered obvious by *Bossen* and *Fleming* for at least the reasons set forth above.

Claim 23 recites:

> leading thread execution circuitry to execute a leading thread of instructions;
>
> trailing thread execution circuitry to execute a trailing thread of instructions;
>
> a memory controller coupled with the leading thread execution circuitry; and
>
> a memory coupled with the leading thread execution circuitry and the trailing thread execution circuitry to store information related to non-deterministic memory access events, wherein the information related to the non-deterministic memory access events is stored at least until a subsequent checkpoint having an associated register architectural state is validated and the non-deterministic memory access events are re-executed if a processing error is detected.

Thus, Applicants claim leading thread execution circuitry, trailing thread execution circuitry where the trailing thread execution circuitry stores information related to non-deterministic memory access events. The information related to non-deterministic memory access events is stored at least until a subsequent checkpoint having associated register state data is validated.

As discussed above, neither *Bossen* nor *Fleming* provide logging of non-deterministic memory access events. Therefore, no combination of *Bossen* and *Fleming* can teach or suggest the invention as claimed in claim 23.

Claims 24-27 depend from claim 23. Because dependent claims include the limitations of the claims from which they depend, Applicants submit that claims 24-27 are not rendered obvious by *Bossen* and *Fleming* for at least the reasons set forth above.

CONCLUSION

For at least the foregoing reasons, Applicants submit that the rejections have been overcome. Therefore, claims 1-27 are in condition for allowance and such action is earnestly solicited. The Examiner is respectfully requested to contact the undersigned by telephone if such contact would further the examination of the present application.

Please charge any shortages and credit any overcharges to our Deposit Account number 02-2666.

Respectfully submitted,
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN, LLP

Date: June 28, 2007

/Paul A. Mendonsa/
Paul A. Mendonsa
Attorney for Applicant
Reg. No. 42,879

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, CA 90025-1026
(503) 439-8778